WPS7849

Policy Research Working Paper 7849

Retooling Poverty Targeting Using Out-of-Sample Validation and Machine Learning

Linden McBride
Austin Nichols

WORLD BANK GROUP
Development Economics Vice Presidency
Operations and Strategy Team
October 2016

Policy Research Working Paper 7849

Abstract

Proxy means test (PMT) poverty targeting tools have become common tools for beneficiary targeting and poverty assessment where full means tests are costly. Currently popular estimation procedures for generating these tools prioritize minimization of in-sample prediction errors; however, the objective in generating such tools is out-of-sample prediction. This paper presents evidence that prioritizing minimal out-of-sample error, identified through cross-validation and stochastic ensemble methods, in PMT tool development can substantially improve the out-of-sample performance of these targeting tools. The USAID poverty assessment tool and base data are used for demonstration of these methods; however, the methods applied in this paper should be considered for PMT and other poverty-targeting tool development more broadly.

This paper is a product of the Operations and Strategy Team, Development Economics Vice Presidency. It is part of a larger effort by the World Bank to provide open access to its research and make a contribution to development policy discussions around the world. Policy Research Working Papers are also posted on the Web at http://econ.worldbank.org.
The authors may be contacted at lem247@cornell.edu.

Retooling Poverty Targeting Using Out-of-Sample Validation and Machine Learning

Linden McBride and Austin Nichols

JEL codes: C140, I320, O220, O150

Linden McBride (corresponding author) is a PhD candidate at Cornell's Dyson School of Applied Economics and Management; her email address is lem247@cornell.edu. Austin Nichols is a principal associate at Abt Associates; his email address is austinnichols@gmail.com. The authors gratefully acknowledge insights from Chris Barrett, Mark Schreiner, Daniel Fink, participants of the University of MN Trade and Development Seminar, participants of the Barrett Research Group Seminar, and two anonymous reviewers. The authors are especially grateful to Nicolai Meinshausen for his innovative quantile regression forest program. All errors are our own.

Accurate targeting is one of the most important components of an effective and efficient food security or social safety net intervention (Barrett and Lentz 2013; Coady, Grosh, and Hoddinott 2004). To achieve accurate targeting, project implementers seek to minimize rates of leakage (benefits reaching those who don't need them) and undercoverage (benefits not reaching those who do need them). Full means tests for identification of project beneficiaries can include detailed expenditure and/or consumption surveys; while effective, such tests are also time consuming and expensive. Proxy means tests (PMTs), a shortcut to full means tests, were first developed for the targeting of social programs in Latin American countries during the 1980s. PMTs have become common tools for targeting and poverty assessment where full means tests are costly (Coady, Grosh, and Hoddinott 2004). Today they are used by USAID (United States Agency for International Development) microenterprise project implementing partners, the World Food Program, and the World Bank, among many others for the purpose of poverty assessment, beneficiary targeting, and program monitoring and evaluation in developing countries (PAT 2014; WBG 2011).

PMT tools are typically developed by assignment of weights, or parameters, to a number of easily verifiable household characteristics via either regression or principal components analysis (PCA) in an available, nationally representative data set. In the regression approach, household-level income/expenditures or poverty status are regressed on household characteristics with the objective of selecting and parameterizing a subset of those characteristics to explain a significant proportion of the variation in expenditures/income or poverty status. In the PCA approach, the parameters are generated by extracting from a set of variables an orthogonal linear combination of a subset of those variables that captures most of the common variation (Filmer and Pritchett 2001; Hastie, Tibshirani, and Friedman 2009). Although each approach has its

advocates, those interested solely in targeting tend to rely on regression approaches, while PCA has become popular among those interested in generating asset indices that may or may not be used for targeting. Note that the problem of developing tools for poverty targeting can be a fundamentally different problem from that of generating asset indices[1]; this paper speaks only to the problem of developing targeting tools.

The regression approach to PMT tool development requires practitioners to select from a large set of potential observables a subset of household characteristics that can account for a substantial amount of the variation in the dependent variable. In practice, this is usually done through stepwise regression and the best performing tool is selected as that which performs best in-sample; more recently, efforts to validate in-sample-generated tools via out-of-sample testing have also been introduced (Schreiner 2006).

Once a PMT tool has been developed from a sample from a particular population, the development practitioner can apply the tool to the subpopulation selected for intervention to rank or classify households according to PMT score. This process involves implementation of a brief household survey in the targeted subpopulation so as to assign values for each of the household characteristics identified during tool development. The observed household characteristics, x_{ij}, are then multiplied by the PMT tool weights, θ_j, for each characteristic j to generate a PMT score for household i, as shown in equation (1):

[1] For example, we might be concerned about endogeneity but not concerned about out-of-sample performance when generating an asset index to estimate the relationship between school enrollment and wealth, as in Filmer and Pritchett (2001). We have no such endogeneity concern when generating targeting tools because we are not attempting causal inference; however, out-of-sample performance is a primary concern.

$$PMTscore_i = \Sigma_j x_{ij}\theta_j. \quad (1)$$

In many applications, the calculated PMT scores are used to rank households from poorest to wealthiest[2] and the poorest households are selected as program beneficiaries. In the case of the USAID poverty assessment tools that will be described below, the use is more conservative: the PMT scores are used to quantify the number of households above and below an identified poverty threshold so as to ensure proper allocation of USAID funds (PAT 2014). The methodological improvements we propose in this paper apply to both types of uses for PMT tools.

Overall, the objective of a PMT tool is to quickly and accurately identify households meeting particular criteria in a new setting (but under the same data-generating process) using a model parameterized with previously available data. Therefore, for PMT tools to serve their purpose, it is important that they perform well not only within the data set or sample in which they were parameterized but also, especially, within the new data set or sample. In other words, high out-of-sample prediction accuracy must be prioritized in the development of PMT tools. In the fields of machine learning and predictive analytics, stochastic ensemble methods have been

[2] There are several long-standing debates as to whether targeting tools, PCA type asset indices, and/or the use of consumption or income data in the regression approach capture long run economic status, permanent income, current consumption levels, current welfare, nonfood spending, or something else altogether. Lee (2014) points out that much of the theoretical support for these various claims is dubious and offers a theoretically grounded approach to the development of asset indices to measure poverty. As much as possible, we remain agnostic on the particular type of well-being that PMT tools capture while noting that the methods we discuss and the way in which we discuss them (e.g., their interpretation as capturing household poverty status) are standard in the literature and in practice.

shown to perform very well out-of-sample due to the bias- and variance-reducing features of such methods.

In this paper, we present evidence that the prioritization of the out-of-sample performance of PMT targeting tools can substantially improve their out-of-sample accuracy. We propose two methods for this prioritization: (1) selecting a tool based on its cross-validation performance and (2) using stochastic ensemble methods, which have cross-validation built in, to develop the tool. Stochastic ensemble methods offer the additional feature, over and above traditional methods combined with cross-validation, of selecting the variables with which to build the tool, an otherwise time-consuming process. We take a set of PMT tools that have been developed by the University of Maryland IRIS Center (IRIS: Institutional Reform and Informal Sector) for the purpose of USAID poverty assessment for demonstration of these methods; however, the methods applied in this paper should be considered for PMT and other poverty targeting tool development more broadly.

We next present the USAID poverty assessment tool development and accuracy evaluation criteria; we then introduce the stochastic ensemble algorithms, regression forests, and quantile regression forests, that we apply to the problem of developing more accurate out-of-sample targeting tools; an explanation of our data and methods follows. We close with results and conclusions.

I. The USAID Poverty Assessment Tool

The development of the USAID poverty assessment tool (PAT) dates from 2000, when the US Congress passed the Microenterprise for Self-Reliance and International Anti-Corruption Act,

mandating that half of all USAID microenterprise funds benefit the very poor (PAT 2014). In the context of this legislation, the very poor are defined as those households living on less than the equivalent of a dollar per day or those households considered "among the poorest 50 percent of households below the country's own national poverty line" (IRIS Center 2005). Subsequent legislation required USAID to develop and certify low-cost tools to enable its microenterprise project-implementing partners[3] to assess the poverty status of microenterprise beneficiaries. USAID engaged the IRIS Center at the University of Maryland in 2003 to create the tools. To date, the IRIS Center has developed, and USAID has certified, tools for 38 countries.[4]

Using existing Living Standards Measurement Study (LSMS) data as well as survey data collected by IRIS, the IRIS Center developed country-specific PAT tools following the general PMT development procedure: they first identified a subset of household characteristics (approximately 15) from the larger data set of 70–125 available observables that accounted for the greatest variation in household level income via an R-squared maximization routine, SAS MAXR[5]; they then selected for the final tool the parameters identified by the statistical model—

[3] The implementing partners who are required to make use of the PAT include "all projects and partner organizations receiving at least US$100,000 from USAID in a fiscal year for microenterprise activities in countries with a USAID-approved tool" (PAT 2014). In 2013, this entailed 71 partners receiving a total of 110 million dollars (USAID MMR).

[4] Albania, Azerbaijan, Bangladesh, Bolivia, Bosnia and Herzegovina, Cambodia, Colombia, East Timor, Ecuador, El Salvador, Ethiopia, Ghana, Guatemala, Haiti, India, Indonesia, Jamaica, Kazakhstan, Kenya, Kosovo, Liberia, Madagascar, Malawi, Mexico, Nepal, Nicaragua, Nigeria, Paraguay, Peru, The Philippines, Rwanda, Senegal, Serbia, Tanzania, Tajikistan, Uganda, Vietnam, and the West Bank.

[5] The MAXR procedure operates by selecting and rejecting variables one by one with the objective of maximizing the improvement in a model's R^2 (SAS 2009).

whether ordinary least squares (OLS), quantile regression, logit, or probit—that produced the highest predictive accuracy in-sample. In some cases, but not all, out-of-sample validation tests were performed.

The predictive ability of the resulting PMT model was evaluated against a number of accuracy criteria—total accuracy, poverty accuracy, undercoverage, leakage, and the balanced poverty accuracy criterion—each of which is defined below. These criteria allow for ex ante evaluation of the generated poverty assessment tools via systematic consideration of each possible outcome/error type as presented in the confusion matrix in table 1: true positive (the true very poor, $p = 1$, are identified by the tool as very poor, $\hat{p} = 1$); false negative (the true very poor, $p = 1$, are identified by the tool as non very poor, $\hat{p} = 0$); false positive (the true non very poor, $p = 0$, are identified by the tool as very poor, $\hat{p} = 1$); true negative (and the true non very poor, $p = 0$, are identified by the tool as non very poor, $\hat{p} = 0$).

The classification literature has developed many metrics based on confusion matrices, such as that presented in table 1, for the assessment of classification accuracy; the IRIS Center draws on standard metrics from the literature and has also developed a new metric for their evaluation of the PAT. Following the IRIS Center and relying on the categories given in table 1, the accuracy criteria we use to assess PAT performance are defined as follows: total accuracy (TA) is the sum of the correctly predicted very poor and the correctly predicted non very poor as a percentage of the total sample, (TA=(TP+TN)/(TP+TN+FP+FN)). Poverty accuracy (PA) is the correctly predicted very poor as a percentage of the total true very poor, (PA=TP/(TP+FN)). The undercoverage rate is the ratio of true very poor incorrectly predicted as non very poor to total true very poor, (UC=FN/(TP+FN)), while the leakage rate is the ratio of true non very poor incorrectly identified as very poor to total true very poor, (LE=FP/(TP+FN)). Finally, the

balanced poverty accuracy criterion (BPAC) is the correctly predicted very poor as a percentage of the true very poor minus the absolute difference between the undercoverage and leakage rates, (BPAC=TP/(TP+FN)-|FN/(TP+FN)-FP/(TP+FN)|). These accuracy criteria are summarized in table 2.

Total accuracy, or one minus mean squared error, is very familiar to economists as a metric for model assessment. However, there are several reasons why total accuracy might not be an adequate metric for assessing the accuracy of a poverty tool. Consider an example wherein a population of 100 includes 10 poor households. A tool that simply classifies the entire population as nonpoor would have a total accuracy rate of 90 percent, which seems quite good. However, this tool would have failed to identify a single poor household. Therefore, metrics beyond total accuracy are necessary for assessment of poverty tool performance; these additional metrics include poverty accuracy (also known as *precision* in the classification and predictive analytics literature) and undercoverage (*false negative*) and leakage (*false positive*) rates. In the example just given, the poverty accuracy of the tool would be 0 percent, and the undercoverage rate would be 100 percent. These additional metrics offer a better picture of the tool's performance than does total accuracy alone. The BPAC combines these three metrics—poverty accuracy, undercoverage, and leakage—by penalizing the poverty accuracy rate with the extent to which the leakage and undercoverage rates exceed one another. The BPAC is an innovation of the IRIS Center; it was created to balance "the stipulations of the Congressional Mandate against the practical implications of the assessment tools" (IRIS 2005). The other criteria are standard in

PMT development. However, it should be noted that IRIS computes leakage in an unconventional manner.[6]

PAT model selection for each country was ultimately made by IRIS based on the BPAC results in-sample. While we follow the prioritization of the BPAC criteria in the analysis that follows, the methods we propose can just as easily be used to meet other prioritized accuracy criteria.

II. Stochastic Ensemble Methods: Regression Forests and Quantile Regression Forests

Classification and regression trees are a class of supervised learning methods that produce predictive models via stratification of a feature (in the case of poverty tool development, a feature is a variable or characteristic) space into a number of regions following a decision rule (Hastie, Tibshirani, and Friedman 2009). A canonical and intuitive example of a classification tree is that of predicting, based on a number of features such as age, gender, and class, who

[6] Whereas leakage rates are commonly computed as FP/(TP+FP), IRIS computes leakage rates as FP/(TP+FN). This adjustment to the denominator in the calculation of leakage rates has two consequences: 1) it can lead to calculated leakage rates that are greater than one, producing a heavy penalty in the calculation of BPAC where leakage occurs (it is not clear that IRIS intended for this outcome); 2) it keeps constant the denominator across poverty accuracy, undercoverage, and leakage rates, allowing IRIS to easily perform the addition and subtraction necessary for the BPAC calculation. We assume this was IRIS's purpose in modifying the denominator.

survived the sinking of the Titanic.[7] While both classification and regression trees can be used to make predictions regarding the poverty status of households based on observable household characteristics, this paper focuses on regression and, in particular, quantile regression forests due to the advantages the latter offers in terms of making predictions about households concentrated at the lower end of the income distribution.

Regression trees operate via a recursive binary splitting algorithm as follows (Hastie, Tibshirani, and Friedman 2009): for *N* observations of response variable, y_i, and a vector of characteristics, $\boldsymbol{x}_{ij}$, where $i = 1,2, \ldots N$ is the number of observations and j=1,2,…*J* is the number of features, consider the splitting variable, x_j, and the split point, where $x_{ij} = s$, that define the half planes R_1 and R_2, as indicated in equation (2):

$$R_1(j,s) = \{x_{ij} | x_{ij} \leq s\} \text{ and } R_2(j,s) = \{x_{ij} | x_{ij} > s\}. \qquad (2)$$

The algorithm selects x_j and s to solve the minimization problem,

$$\min_{js}[\min_{c_1} \Sigma_{x_i \in R_1(j,s)}(y_i - c_1)^2 + \min_{c_2} \Sigma_{x_i \in R_2(j,s)}(y_i - c_2)^2], \qquad (3)$$

where the inner minimizations are solved by

$$c_1 = \frac{1}{n}\Sigma_i(y_i | x_i \in R_1(j,s)) \text{ and } c_2 = \frac{1}{n}\Sigma_i(y_i | x_i \in R_2(j,s)). \qquad (4)$$

In words, the regression tree algorithm chooses the variable, x_j (the splitting variable), and the value of that variable, s (the split point), which minimizes the summed squared distance between the mean response variable and the actual response variables for the observations found in each

[7] See Varian (2014) for an example. Many examples and data are also available at The Comprehensive R Archive Network at http://cran.r-project.org.

of the resulting regions. In this manner, the algorithm effectively weights the response variables by the predictive value of the observations within each region (Lin and Jeon 2006). Once the optimal split in equation (3) is identified, the algorithm proceeds within the new partitions.

One way to think about a regression tree is as an OLS regression for which one knows in advance all of the split variables and split points across which to partition, and then conditionally partition, the feature space, which therefore defines appropriate binary variables and interaction terms to capture these partitions. Such an OLS would return the same results as a regression tree built over the same data. However, such split variables and split points are not known in advance; therefore, what the regression tree algorithm offers over and above an OLS is a heuristic method for the selection of those variables, split points, and conditional splits—the binary variables and their interactions—with which to build the model so as to minimize prediction error. To do this using OLS would require a stepwise regression that iterates and then conditionally iterates through each split point of each variable—a computationally intensive process.

The recursive binary splitting process of the regression tree can continue until a stopping criterion is reached; however, larger trees may overfit the data. In the case that we want to bootstrap over this algorithm—a good idea, as the algorithm may make different splitting decisions in different subsets of the data—it becomes apparent that a bias for variance trade-off

is made as we allow the trees to grow large.[8] A collection of larger trees will have high variance but low bias while a collection of smaller trees will have low variance but high bias.

Fortunately, in this setting, the bias-variance trade-off can be somewhat overcome via a process called bootstrap aggregation, or bagging. Bagging involves bootstrapping a number of approximately unbiased and identically distributed regression trees and then averaging across them so as to reduce the variance of the predictor. However, bagging cannot address the persistent variance that arises due to the fact that the trees themselves are correlated, as they were generated over the same feature space. Consider, for example, a set of B identically distributed but correlated regression trees, each with variance σ^2. If ρ represents the pairwise correlation between the trees, then the variance of the average of these trees is $\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$. As B grows large, the term $\frac{1-\rho}{B}\sigma^2$ will approach zero, reducing the overall variance. However, the first term, $\rho\sigma^2$, persists (Hastie, Tibshirani, and Friedman 2009).

Reducing this persistent variance component of the bagged predictor is the innovation of random forests. Introduced by Breiman (2001), regression forests improve the variance reduction feature of bagged regression trees by decorrelating the trees, and thereby reducing ρ via a random selection of the features (variables) over which the algorithm may split. The number of random features available to the algorithm at any split is typically limited to one-third of the total number of features (Hastie, Tibshirani, and Friedman 2009); this is a tuning parameter of the algorithm.

---

[8] A variety of options for "pruning" trees exist to address these issues in a regression tree framework (Hastie, Tibshirani, and Friedman 2009). We don't discuss these here but move on instead to random forests, which address the problem without pruning.

Critically, in a random forest algorithm, the mean squared error of the prediction is estimated in the "out of bag" sample (OOB), the (on average) third of the training data set on which any given tree has not been built (Breiman 2001), in a manner similar to k-fold cross-validation. This OOB sample offers an unbiased estimate of the model's performance out-of-sample.

The random forest training algorithm produces a collection of B trees, denoted as $\{T(x;\Theta_b)\}_1^B$, where Θ_b indicates the b[th] tree. The regression forest predictor is then the bagged prediction

$$\hat{f}(x_i) = \frac{1}{B}\sum_{b=1}^{B} T(x_i;\Theta_b). \qquad (5)$$

The regression forest algorithm is detailed in the Appendix.

It has been shown that regression forests offer consistent and approximately unbiased estimates of the conditional mean of a response variable (Breiman 2004; Hastie, Tibshirani, and Friedman 2009). However, as elaborated by Koenker (2005), among others, the conditional mean tells only part of the story of the conditional distribution of y given X. Therefore, we also apply quantile regression forests, as developed by Meinshausen (2006), to our PMT tool development.

Meinshausen (2006) draws on insights from Lin and Jeon (2006), who show that random forest predictors can be thought of as weighted means of the response variable, y_i, as shown in equation (6):

$$\hat{f}(x_i) = \frac{1}{B}\sum_{b=1}^{B} T(x_i;\Theta_b) = \sum_{i=1}^{N} \frac{\sum_{b=1}^{B} w_i(x_i;\Theta_b)}{\beta} y_i. \qquad (6)$$

In equation (6), $w_i(x_i;\Theta)$ represents the weight vector obtained by averaging over the observed values in a given region R_l, $(l = 1 \ldots L)$. Application of the weight vector to the response variable

is simply another way of considering the conditional averaging of the response variable, as represented in equation (4) above and shown in equation (7):

$$w_i(x_i;\Theta)y_i = \frac{1}{n}\Sigma_i(y_i|x_i \in R_l(j,s)). \qquad (7)$$

With this insight, Meinshausen (2006) produces quantile regression forests, as a generalization of regression forests in which not only the conditional mean, but the entire conditional distribution of the response variable is estimated (Equation 8):

$$\hat{f}_y(x_i) = \sum_{i=1}^{N} \frac{\sum_{b=1}^{B} w_i(x_i;\Theta_b)}{\beta} 1\{y_i \leq y\}. \qquad (8)$$

Meinshausen (2006) provides a proof for the consistency of this method and demonstrates the gains in predictive performance of quantile regression forests over linear quantile regression. These gains are due to the fact that quantile regression forests retain all the bias-minimizing and variance-reducing components of regression forests in that they bootstrap aggregate across a great number of decorrelated trees; quantile regression forests additionally offer the ability to make predictions across the conditional distribution. A quantile approach is particularly useful for the purposes of PMT tool development due to the fact that the very poor are often concentrated at one end of the conditional income distribution, far from the conditional mean. The quantile regression forest algorithm is detailed in the Appendix.

The advantages that stochastic ensemble methods, such as the regression forest and quantile regression forest algorithms, offer over traditional PMT development tools include the selection of the variables that offer the greatest predictive accuracy without the need to resort to stepwise regression and/or running multiple model specifications—rather, the algorithms build the model—and built-in cross-validation via the out-of-bag error estimates.

Therefore, using regression forest and quantile regression forest algorithms, we expect to realize improvements in the out-of-sample targeting accuracy of the PAT. We note, however, that this method requires the critical assumption that the data-generating process remains unchanged between tool development and tool application. That is, the algorithm can perform well out of sample but not out of population. This limitation plagues any sample-based estimation routine.

III. Empirical Method and Data

We produce a set of country-specific examples from the survey data that was used by the IRIS Center to construct their PATs. We replicate the PAT development process by extracting the same variables that IRIS extracted from the same data sets and then generating identical estimation models. We are limited in our replication process to the use of LSMS data sets that are publicly available. We have additionally constrained ourselves to the LSMS data sets for which income or expenditure aggregates are also publicly available due to the challenges of precisely replicating an income or expenditure aggregate that IRIS may have generated.

From the publicly available data sets meeting these criteria, we selected three nearly arbitrarily: the 2005 Bolivia Encuesta de Hogares (EH), the 2001 Timor Leste Living Standards Survey (TLSS), and the 2004-2005 Malawi Second Integrated Household Survey (IHS2). These data sets present a reasonable representation of the settings in which PATs have been developed. Each data set differs in number of observations, poverty level, and IRIS-selected household characteristics. The data are summarized in table 3, where we can see that the number of household level observations ranges from 1,800 in East Timor to 11,280 in Malawi. Likewise,

the USAID-defined poverty rates range considerably, from 24.2 percent in Bolivia to 64.8 percent in Malawi.

The fourth column of table 3 displays the household-level characteristics selected by IRIS for PAT tool development; many characteristics such as household size, age of household head, household construction materials, and material possessions are common across data sets.

We provide the IRIS reported in-sample accuracy estimates for each country-level data set in each row 1 of Appendix table A1. These are the estimates on which the IRIS model selection was made. We provide the IRIS-reported out-of-sample accuracy assessment results for each country in rows 2–4 of table A1. We replicate the IRIS in-sample models and report the replication estimates in each row 5 of Appendix table A1. Within-country comparisons of our replication estimates (table A1, row 5), with the estimates reported by IRIS (table A1, row 1), serve as a check on how well we have replicated the PAT tool development process. In the case of Bolivia, our replication estimates do not perform as well as those of IRIS; however, it should be noted that IRIS built the Bolivia PAT tool on a randomly selected subset of the data. We cannot replicate precisely the same random draw and so report the full sample estimates. The full sample replication does not perform as well as the half sample performance reported by IRIS, but that half sample is unusual in its high performance, and not representative of the thousand half sample splits we explored or that IRIS reported for their calculation of out-of-sample performance (see rows 2 through 4 of Appendix table A1 for Bolivia). For this reason, we are not concerned about spuriously overestimating the performance of our methods relative to those of IRIS and therefore retain this data set in our analysis. In the case of East Timor and Malawi, our replication estimates are very close to those reported by IRIS, and we are likewise not concerned about unfair comparisons of our methods with those of IRIS.

Our empirical approach is to randomly draw, with replacement, two samples of size N/2 from each country-level data set, producing a training sample and a testing sample. Over this split of the data, we first reproduce IRIS's methods, training their preferred model in the training data and then testing it on 1,000 bootstrap samples of the testing data.[9] However, instead of basing tool selection on in-sample performance as IRIS does, we perform k-fold cross-validation in the training sample and select as our preferred model the one that produces the best BPAC in cross-validation. For this exercise, we use k-fold cross-validation; in particular, we produce 500 iterations of three-fold cross-validation, which entails training the model on two-thirds of the training data set and assessing performance in the remaining third of the training data set on which the model was not trained. We take this approach because it most closely approximates the out-of-bag error produced using the stochastic ensemble methods.

Following the method for out-of-sample testing used by the IRIS center, we test the classification accuracy of the cross-validation-selected tool using 1,000 bootstrapped samples of the testing sample. The out-of-sample performance of this tool in the testing sample is presented for each country in figures 1–3, as well as in Appendix table A1, rows 6 through 8. We refer to this approach of using cross-validation to select the best-performing model in the training sample as the "cross-validation" approach throughout remaining sections to distinguish it from both IRIS's approach and from the stochastic ensemble method approach (note that stochastic ensemble methods also use cross-validation; however, it is referred to as out-of-bag error in that setting).

[9] This method was first used in Schreiner (2006).

We next turn to the stochastic ensemble methods. Over the same split of the data as used for the cross-validation approach, the random forest and quantile regression forest models are built in the training sample where, for any given $(\boldsymbol{x}_i, y_i)$, an average of two-thirds of the training data are used to build bagged regression trees and the remaining third is reserved for out-of-bag, and therefore unbiased, running estimates of the prediction error over a forest of 500 trees.[10] We run the regression forest and quantile regression forest algorithms in R using packages developed by Liaw and Wiener (2002) and Meinshausen (2016), respectively. We select our preferred model as that with the lowest BPAC error in the OOB sample. This model is then taken to the testing sample to assess classification accuracy. The performance of this tool in the testing sample is presented for each country in figures 1–3, as well as in Appendix table A1, rows 9 through 11.

We statistically compare the mean of the IRIS-reported bootstrapped accuracy estimates with those produced using both of our approaches to tool development—the cross-validation approach and the stochastic ensemble approach—using Tukey Kramer tests, selected to account for the family-wise error rate. The results are reported in table 4.

Finally, so as to assess the robustness of our results to the poverty thresholds in each country, we report in Appendix table A2 the performance of our methods as compared with those of IRIS under two new poverty lines: one that is half the original poverty line and a second that is twice the original poverty line. We cannot observe actual IRIS tool performance metrics under

[10] Five hundred trees is the default setting in the randomForest package in R. From casual observation, the OOB error has largely stabilized by the time the forest has reached 200–300 trees; this observation is consistent with the literature (Hastie, Tibshirani, and Friedman 2009).

these new poverty lines, but we estimate the best possible results IRIS could have gotten using their methods and preferred tools by adapting those tools to obtain the greatest BPAC under the new poverty lines. In practice, this means selection of the quantile that offers the best in-sample BPAC under the new poverty lines in Bolivia and Malawi. In the case of East Timor, we include a quantile regression approach along with IRIS's preferred approach under the original poverty line, the probit model, because the probit performs poorly at the lower poverty line. This means we are comparing our cross-validation and ensemble method approaches to the best possible outcomes of the approach employed by IRIS.

IV. Results

Results of the cross-validation (CV) and stochastic ensemble (SE) approaches to PMT tool development are displayed graphically in figures 1, 2, and 3 and numerically in Appendix table A1. In both formats, we compare the out-of-sample bootstrap accuracy estimates of the IRIS-produced tools (rows 2–4 in the table A1) with those produced by each of our approaches. The confidence bars in each figure display the nonparametric bootstrap confidence intervals, where the lower bound is the 2.5th percentile and upper bound is the 97.5th percentile bootstrap estimate. Standard errors are reported in table A1. In addition, Tukey Kramer tests of the differences in the out-of-sample bootstrap means are reported in table 4.

While cross-validation improves on the total accuracy of the IRIS-generated tool only in the case of Bolivia and the stochastic ensemble methods do not improve on the total accuracy at all (figure 1, first graph), gains in poverty accuracy are observed using cross-validation across all countries and using stochastic ensemble methods in both East Timor and Malawi (figure 1,

second graph). Recall from the discussion above that total accuracy has serious limitations as a metric for assessing the performance of a poverty-targeting tool.

From figure 2 (first graph), we can see that these gains in poverty accuracy are not without trade-offs: the leakage rates for the cross-validation and stochastic ensemble approaches are significantly greater than those reported for the IRIS-generated tools in both Bolivia and East Timor, meaning that these tools err on the side of classifying nonpoor households as poor. Given that leakage rates are heavily penalized by the IRIS accuracy metrics, these increases are not very surprising. Meanwhile, the cross-validation approach performs much better than IRIS's in terms of undercoverage rates; the undercoverage rate is decreased across all countries (figure 2, second graph). The stochastic ensemble approach likewise outperforms IRIS's in both East Timor and Malawi.

The critical question, then, is how these trade-offs net out in terms of USAID's key accuracy metric, the BPAC. Figure 3 demonstrates that the accuracy of the cross-validation approach outperforms that of the IRIS-generated tool in each country. Improvements range from 2.7 percent in Malawi to 17.5 percent in Bolivia. The performance of the stochastic ensemble approach closely follows that of the cross-validation approach in both East Timor and Malawi; although the cross-validation results are statistically significantly different from the stochastic ensemble results, the magnitude of those differences is trivial in the case of Malawi and quite small in the case of East Timor (table 4).

In addition to gains in average BPAC, we also see large gains in the lower bound (2.5th percentile) performance using cross-validation and stochastic ensemble methods. The cross-validation (stochastic ensemble) approach improves the lower bound BPAC accuracy in Boliva by 38 (7) percent, in East Timor by 11 (8) percent, and in Malawi by 3 (2) percent.

Although the gains in poverty accuracy and BPAC in Malawi using the cross-validation approach are not as impressive as those in Bolivia and East Timor, note that the tool is able to outperform the already relatively accurate IRIS tool for Malawi in terms of these metrics while also reducing *both* the leakage and undercoverage rates.

The relatively strong performance of the cross-validation approach compared with the stochastic ensemble approach is due to the fact that the cross-validation approach benefits from IRIS's time and effort in selecting from a large set of possible variables a subset that explains much of the variation in the dependent variable. Because we have limited our analysis to the same subset of variables as selected by IRIS for their preferred models, the relative strengths of the stochastic ensemble methods in terms of variable selection are not well displayed through this analysis. Therefore, it remains an open question (that we plan to address in a later paper) as to whether our stochastic ensemble approach would outperform the combination of IRIS's parametric model with cross-validation had we begun with the full set of 70–125 variables instead of the selected subset. Our analysis does suggest, however, that the proxy means test tool developer who prefers to skip the time-consuming and computationally intensive process of stepwise regression followed by the comparison of multiple model specifications would do at least nearly as well in terms of out-of-sample performance as the tool developer who does take the time to perform these analyses and then combine them with cross-validation.

Finally, the robustness results for the assessment of tool performance under new poverty lines are reported in appendix table A2. From a comparison of rows 2, 6, and 9 for each country, we can see that the cross-validation and stochastic ensemble approaches perform about the same as the IRIS approach under the new poverty lines. Overall, however, across all results, including the robustness results, we find that the cross-validation and stochastic ensemble approaches do

no worse than, and in many cases substantially outperform, the traditional approach to PMT tool development.

V. Conclusion

We have proposed methods for the improvement of a particular type of poverty-targeting tool: proxy means test targeting. In the country-level case studies analyzed here, prioritization of the out-of-sample performance of these targeting tools during tool development either through selecting a model based on its cross-validation performance or using a method such as stochastic ensemble methods that both selects variables and performs cross-validation along the way can significantly improve the out-of-sample performance of these tools. In particular, we find that application of cross-validation and stochastic ensemble methods to the problem of developing a poverty-targeting tool produces a gain in poverty accuracy, a reduction in undercoverage rates, and an overall improvement in BPAC in comparison to traditional methods.

Our analysis takes as given the IRIS-selected PAT variables so as to demonstrate the power of machine learning methods in this setting; however, beginning with a larger set of variables over which the stochastic ensemble methods may build a targeting model may produce even greater gains in targeting accuracy for this approach than observed here.[11] Therefore, the gains in accuracy we have reported are likely conservative. Moreover, applying a stochastic ensemble approach over a larger set of variables would obviate the time-consuming tasks of both

[11] Note, however, that an algorithm cannot be given completely free range in variable selection as the selected variables must be easily observable household characteristics that can be quickly verified with a visit to the household for them to contribute meaningfully to a PMT test.

stepwise regression for variable selection and the process of running and comparing the performance of multiple statistical models, as was done by the IRIS center. Overall, our findings suggest that further exploration of machine learning methods for PMT tool development is merited.

VI. Appendix

Random forest algorithm (Hastie, Tibshirani, and Friedman 2009; Breiman 2001):

1. Grow B trees, $T(\theta_b), b = 1, \dots, B$, by recursively repeating steps (a)-(c):
    a. Select *m* variables at random from the total J variables (j=1,…J).
    b. Select variable x_j and split point $x_{ij} = s$ to solve the minimization problem as shown in EQ2–EQ4.
    c. Split data into the resulting regions.
2. Output ensemble of trees $\{T_b\}_1^B$.
3. To make prediction at new point, *x*, drop observation down all trees and calculate $\hat{f}_{rf}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

Quantile regression forest algorithm (Meinshausen 2006):

1) Grow B trees, $T(\theta_b), b = 1, \dots, B$, as in the random forests algorithm. However, retain the value of all observation in a given region, not just their average.

2) For a given x_{ij}, drop observation down all trees and compute the weight, $w_i(x_i; \Theta_b)$, of observation i for every tree, b, as $w_i(x_i; \Theta_b) = \frac{1\{x_i \in R_l(j,s)\}}{\sum_{i=1}^{n} 1\{x_i \in R_l(j,s)\}}$. Then compute the weight for every observation as an average over all trees as $\frac{\sum_{b=1}^{B} w_i(x_i;\Theta_b)}{\beta}$.

3) Compute the estimate of the distribution function as $\sum_{i=1}^{N} \frac{\sum_{b=1}^{B} w_i(x_i;\Theta_b)}{\beta} 1\{y_i \leq y\}$ for all y.

REFERENCES

Barrett, C. B., and E. Lentz. 2013. "Hunger and Food Insecurity." In D. Brady and L.M. Burton, eds., *The Oxford Handbook of Poverty and Society.* Oxford: Oxford University Press.

Breiman, L. 2001. "Random Forests." *Machine Learning* 45: 5–32.

———. 2004. "Consistency for a Simple Model of Random Forests." Technical Report. University of California-Berkeley.

Coady, D., M. Grosh, and J. Hoddinott. 2004. *Targeting of Transfers in Developing Countries: Review of Lessons and Experience*. Washington, DC: The International Bank for Reconstruction and Development.

Filmer, D., and L. H. Pritchett. 2001. "Estimating Wealth Effects without Expenditure Data or Tears: An Application to Educational Enrollments in States of India." *Demography* 38 (1): 115–32.

Grosh, M., and J. Baker. 1995. "Proxy Means Tests for Targeting Social Programs." LSMS Working Paper No. 118. The World Bank, Washington, DC.

Hastie, T., R. J. Tibshirani, and J. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2nd ed. New York: Springer.

IRIS Center. 2005. "Note on Assessment and Improvement of Tool Accuracy. Poverty Assessment Tools." USAID. Accessed January 2014. http://www.povertytools.org/training_documents /Introduction%20to%20PA/Accuracy_Note.pdf.

———. 2007. "Poverty Assessment Tool Accuracy Submission. USAID/IRIS Tool for Timor-Leste. Poverty Assessment Tools." USAID. Accessed January 2014. http://www.povertytools.org/tools.html.

———. 2009. "Poverty Assessment Tool Accuracy Submission. USAID/IRIS Tool for Bolivia. Poverty Assessment Tools." USAID. Accessed January 2014. http://www.povertytools.org/tools.html.

———. 2012. "Poverty Assessment Tool Accuracy Submission. USAID/IRIS Tool for Malawi. Poverty Assessment Tools." USAID. Accessed January 2014. http://www.povertytools.org/tools.html.

Koenker, R. 2005. *Quantile Regression*. Cambridge: Cambridge University Press.

Liaw, A., and M. Wiener. 2002. "Classification and regression by randomForest." *R News* 2:18–22.

Lee, D. 2014. "Measuring Poverty Using Asset Ownership: Developing a Theory-Driven Asset Index Incorporating Utility and Prices." Unpublished Job Market Paper. University of California-Berkeley. Accessed January 2014. http://areweb.berkeley.edu/candidate/Diana_Lee.

Lin, Y., and Y. Jeon. 2006. “Random Forest and Adaptive Nearest Neighbors.” *Journal of the American Statistical Association* 101 (474): 578–590.

Meinshausen, N. 2006. “Quantile Regression Forests.” *Journal of Machine Learning Research* 7: 983–99.

Meinshausen, N. 2016. quantregForest: Quantile Regression Forests. R package version 1.3-5. Available at http://CRAN.R-project.org/package=quantregForest

PAT (Poverty Assessment Tool). 2014. “Quantifying the Very Poor. Poverty Assessment Tools Website.” Accessed February 2014. http://www.povertytools.org.

R Development Core Team. 2005. “R: A language and environment for statistical computing.” R Foundation for Statistical Computing. Vienna, Austria.

SAS Institute Inc. 2009. “SAS/STAT 9.2 User’s Guide, Second Edition.” SAS Institute Inc., Cary, NC. Accessed May 13, 2012. https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#titlepage.htm.

Schreiner, M. 2006. “A Simple Poverty Scorecard for Bangladesh, Report to Grameen Foundation USA.” Working Paper. Accessed 15 February 2016. www.microfinance.com/English/Papers/Scoring_Poverty_Bangladesh.pdf. Saint Louis, MO: Microfinance Risk Management, L.L.C.

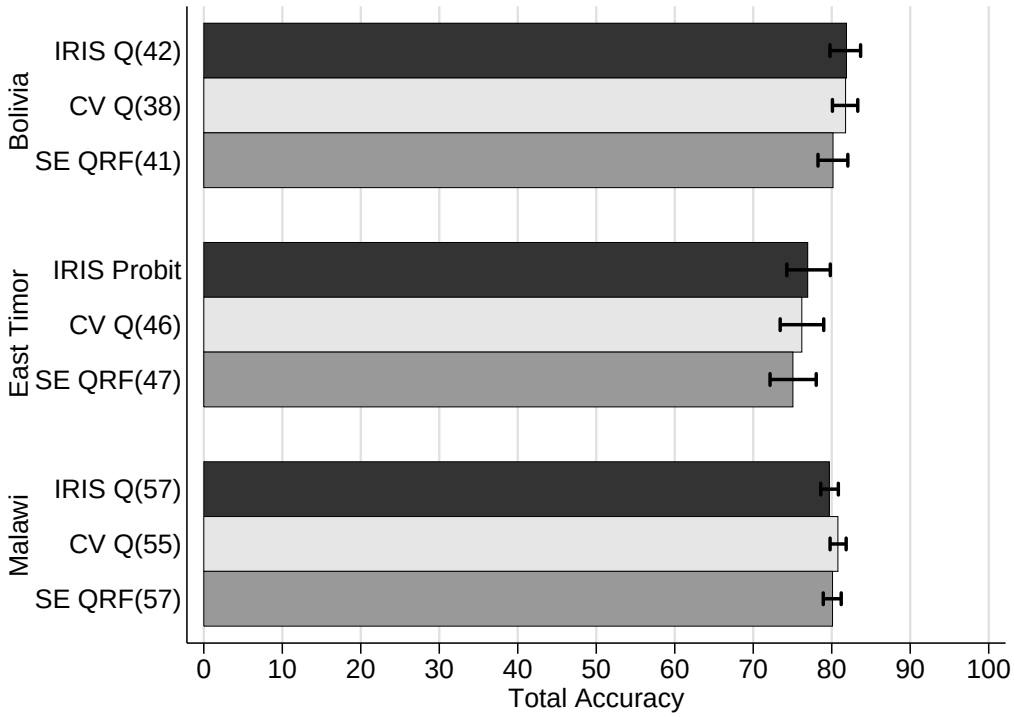
USAID MRR. “USAID Microenterprise Results Reporting Portal.” Accessed December 17, 2014. eads.usaid.gov/mrr/.

Varian, H. 2014. “Big Data: New Tricks for Econometrics.” *Journal of Economic Perspectives* 28 (2): 3–28.

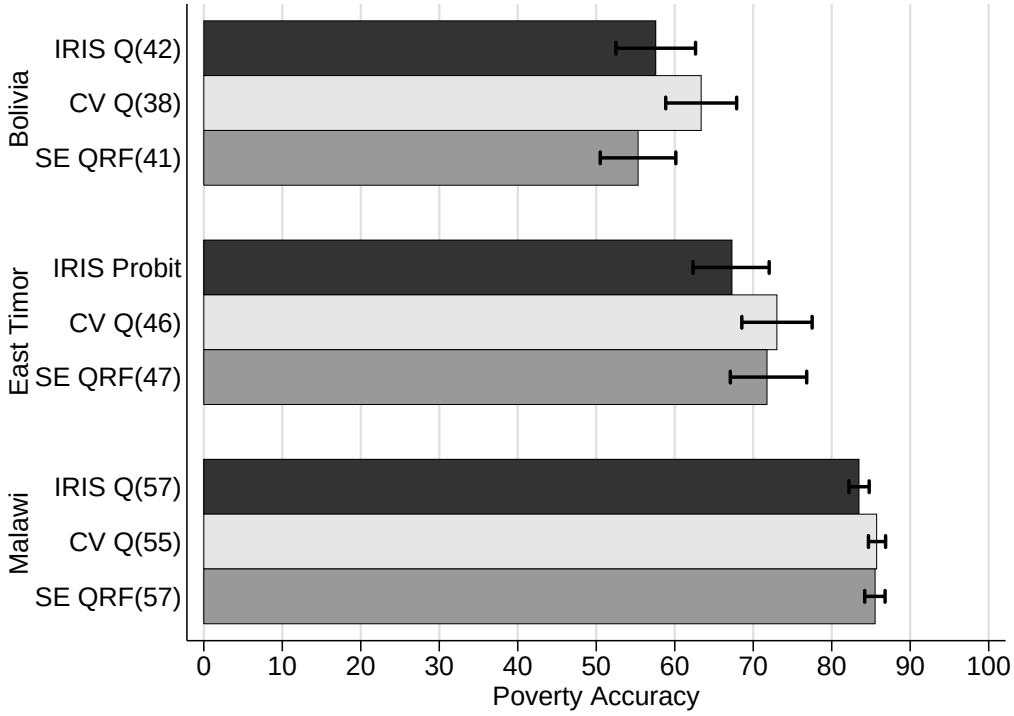
WBG (World Bank Group). 2011. *Targeting: Safety Nets and Transfers: Proxy Means Testing.* Washington, DC: The World Bank. Accessed May 2014. http://web.worldbank.org/WBSITE/EXTERNAL/TOPICS/EXTSOCIALPROTECTION/EXTSAFETYNETSANDTRANSFERS/0,,contentMDK:22188486~pagePK:210058~piPK:210062~theSitePK:282761,00.html

Figure 1. Total and Poverty Accuracy by Country and Estimation Procedure

(a)



(b)

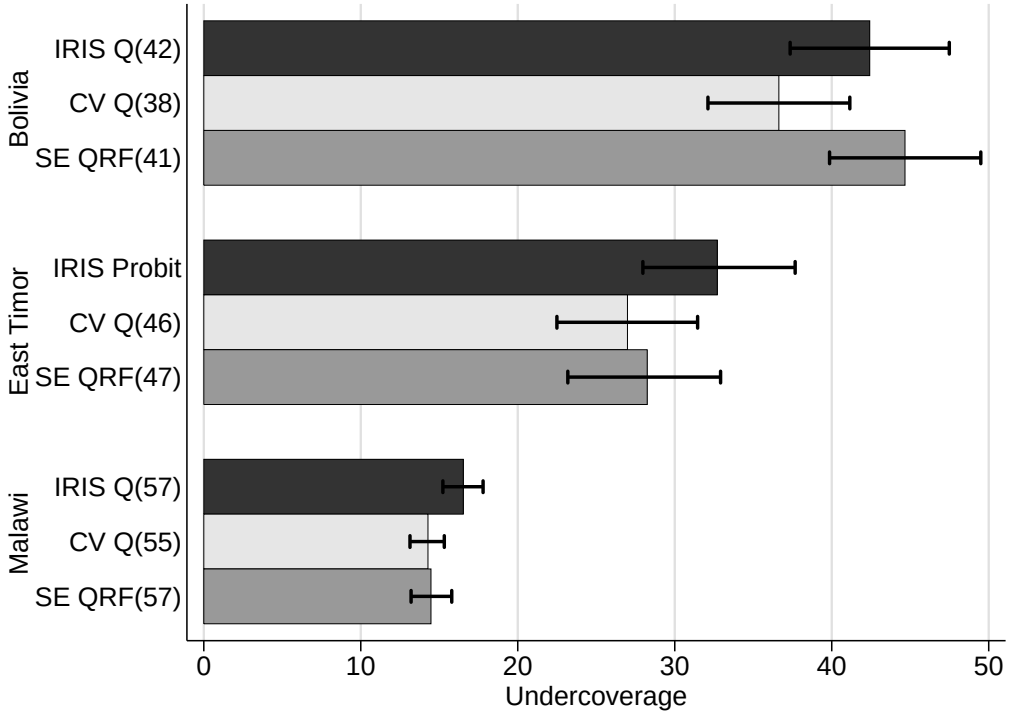


Notes: "IRIS Q(#)" indicates quantile regression (Q) estimated by IRIS at the #th quantile. "CV Q(#)" indicates quantile regression estimated by the authors using cross-validation (CV) at the #th quantile. "SE QRF(#)" indicates quantile regression forest (QRF) estimated by the authors using stochastic ensemble methods (SE) at the #th quantile. "IRIS probit" indicates probit regression estimated by IRIS. Error bars reflect the nonparametric confidence intervals.

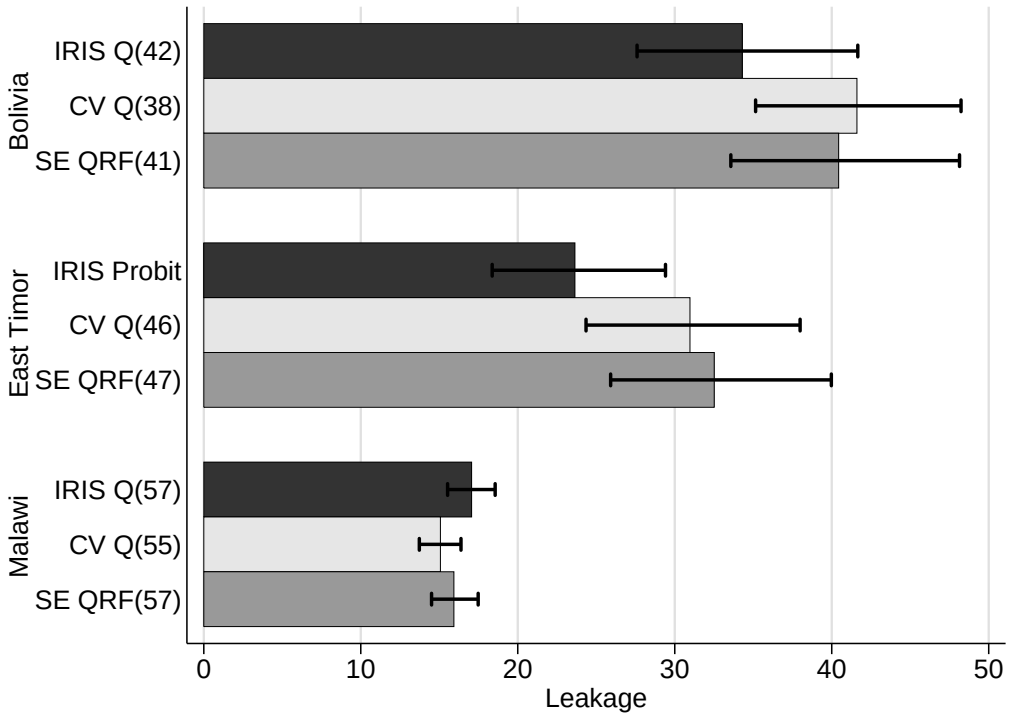
Source: Authors' and IRIS center's estimates using data and procedures detailed in the text.

Figure 2. Leakage and Undercoverage Rates by Country and Estimation Procedure

(a)



(b)

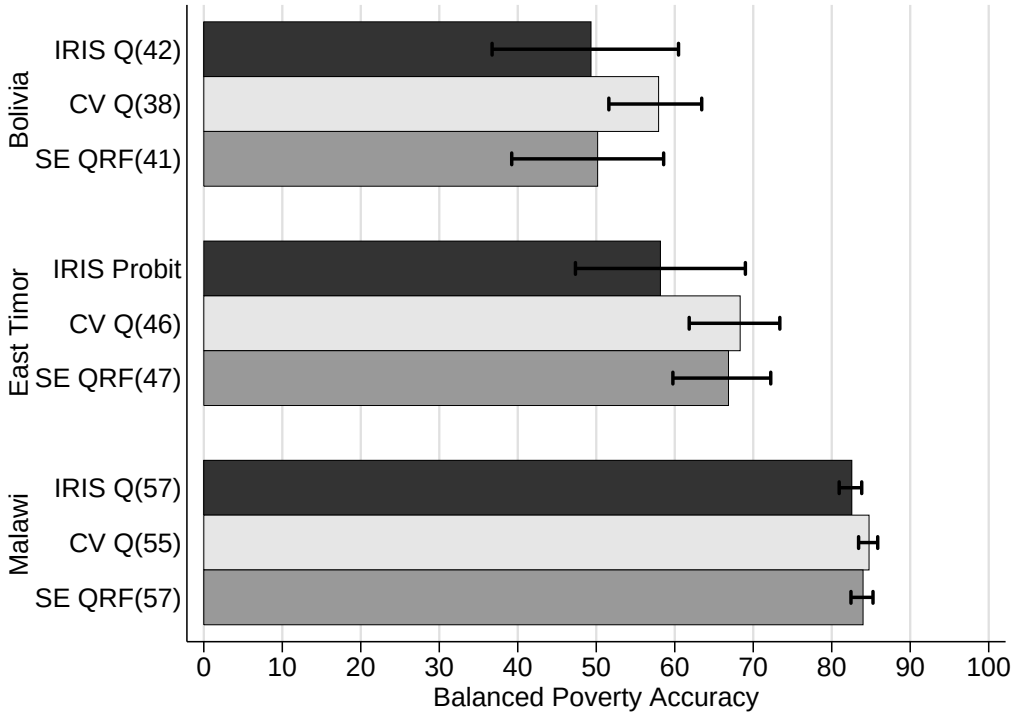


Notes: "IRIS Q(#)" indicates quantile regression (Q) estimated by IRIS at the #th quantile. "CV Q(#)" indicates quantile regression estimated by the authors using cross-validation (CV) at the

#th quantile. “SE QRF(#)” indicates quantile regression forest (QRF) estimated by the authors using stochastic ensemble methods (SE) at the #th quantile. “IRIS probit” indicates probit regression estimated by IRIS. Error bars reflect the nonparametric confidence intervals.

Source: Authors’ and IRIS center’s estimates using data and procedures detailed in the text.

Figure 3. Balanced Poverty Accuracy Criteria by Country and Estimation Procedure



Notes: “IRIS Q(#)” indicates quantile regression (Q) estimated by IRIS at the #th quantile. “CV Q(#)” indicates quantile regression estimated by the authors using cross-validation (CV) at the #th quantile. “SE QRF(#)” indicates quantile regression forest (QRF) estimated by the authors using stochastic ensemble methods (SE) at the #th quantile. “IRIS probit” indicates probit regression estimated by IRIS. Error bars reflect the nonparametric confidence intervals.

Source: Authors’ and IRIS center’s estimates using data and procedures detailed in the text.

Table 1. Poverty Prediction Outcomes

	$p = 1$	$p = 0$
$\hat{p} = 1$	True positive (TP)	False positive (FP)
$\hat{p} = 0$	False negative (FN)	True negative (TN)

Source: Standard confusion matrix.

Table 2. Targeting Accuracy Metrics

Total accuracy	TA=(TP+TN)/(TP+TN+FP+FN)
Poverty accuracy	PA=TP/(TP+FN)
Leakage	LE=FP/(TP+FN)
Undercoverage	UC=FN/(TP+FN)
Balanced poverty accuracy criterion	BPAC=TP/(TP+FN)-|FN/(TP+FN)-FP/(TP+FN)|

Source: Authors' summary based on IRIS Center 2005.

Table 3. LSMS Surveys and Variables Used in PAT Development and Replicated by Authors

County	Data	Obs.	IRIS selected variables	Poverty rate (%)
Bolivia	2005 Encuesta de Hogares (EH)	4,086	hhsize, hhsize2, age head, age head2, regions, rural, sublease, brick wall, wood wall, dirt floor, cement floor, fridge, radio, tv, dvd, fan, car, number beds, number kitchens, number computers, sheep	24.03
Malawi	2004-2005 Second Integrated Household Survey (IHS2)	11,280	hhsize, hhsize2, age head, age head2, regions, rural, never married, share of adults with out education, share of adults who can read, number of rooms, cement floor, electricity, flush toilet, soap, bed, bike, music player, coffee table, iron, garden, goats	64.78
East Timor	2001 Timor Leste Living Standards Survey (TLSS)	1,800	hhsize, hhsize2, age head, age head2, regions, rattantin wall, leaf roof, concreter or tile roof, number rooms, private water, shared water, toilet is a bowl or bucket, electricity light, private light, fan, number of adults who read, farmland, number of axes number of baskets, number of chickens	44.73

Source: Authors' summary based on the data indicated as well as reports from IRIS Center 2007, 2009, and 2012.

Table 4. Tukey-Kramer Tests of Equality of Bootstrap Poverty Accuracy and BPAC Means across Estimates

		Poverty accuracy		Balanced poverty accuracy criteria	
	Estimation	Difference	TK test statistic	Difference	TK test statistic
Bolivia	CV vs IRIS	5.79*	37.55	8.61*	28.20
	SE vs IRIS	-2.25*	-14.07	0.85	2.38
	CV vs SE	8.04*	54.14	7.76*	29.04
East Timor	CV vs IRIS	3.69*	23.89	2.78*	11.87
	SE vs IRIS	2.43*	15.43	1.29*	5.39
	CV vs SE	1.26*	8.40	1.49*	7.68
Malawi	CV vs IRIS	2.25*	59.06	2.19*	50.03
	SE vs IRIS	2.06*	49.11	1.43*	30.85
	CV vs SE	0.19	4.90	0.76*	17.51

Note: CV = cross-validation estimates; IRIS = IRIS reported estimates; SE = stochastic ensemble estimates.

* Indicates difference is significant at 1% significance level.

Source: Authors' estimates using data and procedures detailed in the text.

Table A1. A Comparison of IRIS, Cross-Validation, and Stochastic Ensemble Accuracy Results

Country	Source	Estimation	TA	PA	UC	LE	BPAC
Bolivia (2005 EH)	IRIS	1) QR (0.42)-In sample (half)	83.65	67.18	32.82	33.29	66.71
		2) QR (0.42)[a]	81.88	57.58	42.42	34.3	49.33
		3) Std. err.	1.02	2.61	2.61	3.6	6.11
		4) QR (0.42)[b]	[79.78, 83.68]	[52.51, 62.65]	[37.35, 47.49]	[27.6, 41.66]	[36.73, 60.48]
	Rep	5) QR (0.42) rep.-In sample (full)	82.45	60.69	39.30	33.71	55.10
	Cross-validation	6) QR (0.38)[a]	81.76	63.37	36.63	41.61	57.94
		7) Std. err.	0.86	2.25	2.25	3.39	3.04
		8) QR (0.38)[b]	[80.10, 83.32]	[58.84, 67.88]	[32.12,41.15]	[35.15, 48.24]	[50.61, 63.44]
	Stochastic ensemble	9) QRF (0.41)[a]	80.17	55.33	44.67	40.44	50.18
		10) Std. err.	0.95	2.44	2.44	3.78	5.14
		11) QRF (0.41)[b]	[78.25, 82.03]	[50.51, 60.12]	[39.88, 49.49]	[33.59, 48.11]	[39.26, 58.57]
East Timor (2001 TLSS)	IRIS	1) Probit-In sample (full)	77.14	75.08	24.92	26.20	73.79
		2) Probit[a,c]	75.56	69.32	30.68	28.71	65.56
		3) Std. err.	1.52	2.56	2.56	3.38	4.33
		4) Probit[b,c]	[72.63, 78.50]	[64.38, 74.33]	[25.67, 35.62]	[22.40, 35.58]	[55.57, 72.08]
	Rep	5) Probit rep.-In sample (full)	77.16	71.41	28.59	27.63	70.45
	Cross-validation	6) QR (0.46)[a]	76.19	73.01	26.99	30.97	68.34
		7) Std. err.	1.42	2.32	2.32	3.33	2.96
		8) QR (0.46)[b]	[73.43, 77.51]	[73.43, 77.51]	[22.49, 31.46]	[24.35, 37.99]	[61.84, 73.38]
	Stochastic ensemble	9) QRF (0.47)[a]	75.05	71.75	28.25	32.51	66.85
		10) Std. err.	1.50	2.42	2.42	3.55	3.17
		11) QRF (0.47)[b]	[72.19, 78.03]	[67.12, 76.72]	[23.28, 32.88]	[25.94, 39.90]	[59.77, 72.22]

Malawi (2004/5 IHS2)	IRIS	1) QR (0.57)-In sample (half)	80.15	84.12	15.88	16.43	83.57
		2) QR (0.57)[a]	79.69	83.47	16.53	17.06	82.56
		3) Std. err.	0.55	0.65	0.65	0.76	0.74
		4) QR (0.57)[b]	[78.6, 80.84]	[82.2, 84.77]	15.23, 17.79]	[15.53, 18.56]	[80.95, 83.82]
	Rep	5) QR (0.57) rep.-In sample (full)	80.82	84.88	15.11	14.39	84.17
	Cross-validation	6) QR (0.55)[a]	80.79	85.72	14.28	15.07	84.75
		7) Std. err.	0.52	0.55	0.55	0.69	0.64
		8) QR (0.55)[b]	[79.79, 81.84]	[84.68, 86.86]	[13.14, 15.32]	[13.73, 16.38]	[83.42, 85.86]
	Stochastic ensemble	9) QRF (0.57)[a]	80.10	85.53	14.47	15.93	83.99
		10) Std. err.	0.58	0.67	0.67	0.75	0.73
		11) QRF (0.57)[b]	[78.93, 81.19]	[84.22, 86.80]	[13.20, 15.78]	[14.51, 17.47]	[82.46, 85.25]

Note: QR(#) = quantile regression estimated at the #th quantile; QRF(#) = quantile regression forest estimated at the #th quantile.

[a]Bootstrapped 1,000 times, with replacement, mean reported.

[b]Bootstrapped 1,000 times, with replacement; 95% bootstrap confidence interval reported, where lower bound is 2.5% and upper bound is 97.5%.

[c]Because these bootstrapped estimates were not available in materials made public by IRIS, the estimates reported here were calculated by the authors based on the replication sample and model.

Source: Authors' and IRIS center's estimates using data and procedures detailed in the text.

Table A2. A Comparison of IRIS, Cross-Validation, and Stochastic Ensemble Accuracy Results under Halved and Doubled Poverty Lines

Data		Estimation	TA	PA	UC	LE	BPAC	Poverty line	Poverty rate (%)
Bolivia (2005 EH)	IRIS	2) QR (0.22)[a]	94.55	41.65	58.35	65.89	30.07		
		3) Std. err.	0.54	5.45	5.45	11.72	9.53		
		4) QR (0.22)[b]	[93.44, 95.56]	[30.72, 52.63]	[47.37, 69.28]	[45.54, 92.19]	[6.73, 44.38]		
	Cross-validation	6) QR(0.24)[a]	94.53	41.20	58.80	66.31	29.65		
		7) Std. err.	0.56	5.44	5.44	11.85	9.50	Half	4.92
		8) QR (0.22\4)[b]	[93.39, 95.61]	[31.14, 52.47]	[47.53, 68.86]	[44.93, 91.90]	[7.90, 44.58]		
	Stochastic ensemble	9) QRF (0.26)[a]	94.39	43.65	56.35	71.03	26.94		
		10) Std. err.	0.56	5.71	5.71	13.25	11.70		
		11) QRF (0.26)[b]	[93.24, 95.43]	[32.00, 55.00]	[45.00, 68.00]	[47.66, 100.00]	[0.00, 45.27]		
	IRIS	2) QR (0.54)[a]	78.90	82.64	17.36	16.65	81.10		
		3) Std. err.	0.94	1.12	1.12	1.35	1.86		
		4) QR (0.54)[b]	[77.11, 84.79]	[80.40, 84.79]	[15.21, 19.60]	[14.14, 19.17]	[76.61, 83.88]		
	Cross-validation	6) QR (0.52)[a]	79.01	83.60	16.40	17.45	81.92		
		7) Std. err.	0.94	1.11	1.11	1.39	1.31	Double	62.26
		8) QR (0.52)[b]	[77.17, 80.83]	[81.38, 85.67]	[14.33, 18.62]	[14.84, 20.09]	[79.05, 84.14]		
	Stochastic ensemble	9) QRF (0.54)[a]	77.89	83.66	16.34	19.32	80.62		
		10) Std. err.	1.02	1.14	1.14	1.38	1.33		
		11) QRF (0.54)[b]	[75.99, 79.79]	[81.41, 85.77]	[14.22, 18.59]	[16.68, 21.99]	[78.01, 83.10]		

East Timor (2001 TLSS)	IRIS	2) Probit[a]	90.82	28.51	71.50	23.37	-19.62	Half	10.65
		3) Std. err.	1.03	5.30	5.30	6.03	12.06		
		4) Probit[b]	[88.69, 92.75]	[18.79, 39.13]	[60.87, 81.21]	[13.37, 36.72]	[-42.95, 3.82]		
		2) QR (0.27)[a]	89.02	49.26	50.74	62.58	35.45		
		3) Std. err.	1.11	5.91	5.91	10.95	9.64		
		4) QR (0.27)[b]	[86.81, 91.25]	[38.03, 61.41]	[38.59, 61.97]	[43.63, 85.61]	[14.04, 51.27]		
	Cross-validation	6) QR (0.28)[a]	88.76	46.09	53.91	61.67	35.04		
		7) Std. err.	1.05	5.44	5.43	10.50	8.71		
		8) QR (0.28)[b]	[86.71, 90.81]	[35.29, 56.88]	[43.12, 64.71]	[42.44, 83.23]	[16.26, 48.94]		
	Stochastic ensemble	9) QRF (0.28)[a]	89.34	39.20	60.80	48.97	23.91		
		10) Std. err.	1.20	5.80	5.80	11.73	13.89		
		11) QRF (0.28)[b]	[86.99, 91.70]	[27.77, 50.55]	[49.45, 72.23]	[29.46, 74.37]	[-5.68, 45.75]		
	IRIS	2) Probit[a]	84.15	93.04	6.96	13.72	86.28	Double	80.20
		3) Std. err.	1.08	0.67	0.67	1.37	1.37		
		4) Probit[b]	[82.75, 85.75]	[92.20, 94.07]	[5.93, 7.80]	[12.12, 15.51]	[84.49, 87.88]		
		2) QR (0.60)[a]	83.34	89.27	10.73	11.16	87.72		
		3) Std. err.	1.33	1.27	1.27	1.43	1.61		
		4) QR (0.60)[b]	[80.75, 85.75]	[86.70, 91.68]	[8.32, 13.30]	[8.33, 14.04]	[83.82, 90.33]		
	Cross-validation	6) QR (0.57)[a]	83.86	91.18	8.82	12.40	87.58		
		7) Std. err.	1.21	1.06	1.06	1.44	1.41		
		8) QR (0.57)[b]	[81.61, 86.10]	[89.16, 93.30]	[6.70, 10.84]	[9.65, 15.33]	[84.67, 90.17]		
	Stochastic ensemble	9) QRF (0.58)[a]	82.63	89.96	11.04	11.78	87.28		
		10) Std. err.	1.28	1.26	1.26	1.44	1.46		
		11) QRF (0.58)[b]	[80.04, 85.09]	[86.52, 91.29]	[8.71, 13.48]	[8.99, 14.59]	[84.00, 89.58]		

Malawi (2004/5 IHS2)	IRIS	2) QR (0.41)[a]	79.67	58.19	41.81	45.48	54.25		
		3) Std. err.	0.56	1.15	1.48	2.42	2.17		
		4) QR (0.41)[b]	[78.58, 80.64]	[55.38, 61.15]	[38.85, 44.62]	[40.77, 50.31]	[49.63, 58.19]		
	Cross-validation	6) QR (0.40)[a]	79.59	56.97	40.02	47.48	52.52		
		7) Std. err.	0.56	1.36	1.36	2.40	2.40	Half	23.43
		8) QR (0.40)[b]	[78.54, 80.67]	[57.21, 62.88]	[37.11, 42.79]	[43.06, 52.16]	[47.84, 56.91]		
	Stochastic ensemble	9) QRF (0.42)[a]	79.24	56.04	43.96	45.15	53.43		
		10) Std. err.	0.58	1.56	1.56	2.41	2.13		
		11) QRF (0.42)[b]	[78.09, 80.32]	[53.04, 59.10]	[40.91, 46.96]	[40.63, 49.76]	[48.74, 57.10]		
	IRIS	2) QR (0.66)[a]	92.12	95.95	4.05	4.65	95.32		
		3) Std. err.	0.38	0.29	0.29	0.33	0.31		
		4) QR (0.66)[b]	[91.37, 92.86]	[95.36, 96.51]	[3.50, 4.64]	[4.05, 5.32]	[94.67, 95.88]		
	Cross-validation	6) QR (0.64)[a]	92.34	96.26	3.74	4.72	95.28		
		7) Std. err.	0.35	0.27	0.27	0.31	0.30	Double	90.65
		8) QR (0.64)[b]	[91.63, 93.01]	[95.75, 96.76]	[3.24, 4.24]	[4.14, 5.36]	[94.64, 95.84]		
	Stochastic ensemble	9) QRF (0.66)[a]	92.11	95.76	4.23	4.48	95.36		
		10) Std. err.	0.37	0.30	0.30	0.31	0.33		
		11) QRF (0.66)[b]	[91.33, 92.81]	[95.17, 96.33]	[3.82, 5.0]	[3.89, 5.11]	[94.62, 95.91]		

Note: QR(#) = quantile regression estimated at the #th quintile; QRF(#) = quantile regression forest estimated at the #th quantile.

[a]Bootstrapped 1,000 times, with replacement, mean reported.

[b]Bootstrapped 1,000 times, with replacement; 95% bootstrap confidence interval reported, where lower bound is 2.5% and upper bound is 97.5%.

Source: Authors' and IRIS center's estimates using data and procedures detailed in the text.